UNITED STATES PATENT AND TRADEMARK OFFICE

UNITED STATES DEPARTMENT OF COMMERCE
United States Patent and Trademark Office
Address: COMMISSIONER FOR PATENTS
P.O. Box 1450
Alexandria, Virginia 22313-1450
www.uspto.gov

APPLICATION NO.	FILING DATE	FIRST NAMED INVENTOR	ATTORNEY DOCKET NO.	CONFIRMATION NO.
10/789,311	02/27/2004	Sheueling Chang Shantz	6000-31500	9201

58467 7590 07/14/2009

MHKKG/SUN
P.O. BOX 398
AUSTIN, TX 78767

EXAMINER	
JOHNSON, CARLTON	

ART UNIT	PAPER NUMBER
2436	

NOTIFICATION DATE	DELIVERY MODE
07/14/2009	ELECTRONIC

Please find below and/or attached an Office communication concerning this application or proceeding.

The time period for reply, if any, is set in the attached communication.

Notice of the Office communication was sent electronically on above-indicated "Notification Date" to the following e-mail address(es):

patent_docketing@intprop.com
ptomhkkg@gmail.com

Office Action Summary	Application No.	Applicant(s)	
	10/789,311	SHANTZ ET AL.	
	Examiner	Art Unit	
	CARLTON V. JOHNSON	2436	

-- The MAILING DATE of this communication appears on the cover sheet with the correspondence address --

Period for Reply

A SHORTENED STATUTORY PERIOD FOR REPLY IS SET TO EXPIRE *3* MONTH(S) OR THIRTY (30) DAYS, WHICHEVER IS LONGER, FROM THE MAILING DATE OF THIS COMMUNICATION.

- Extensions of time may be available under the provisions of 37 CFR 1.136(a). In no event, however, may a reply be timely filed after SIX (6) MONTHS from the mailing date of this communication.
- If NO period for reply is specified above, the maximum statutory period will apply and will expire SIX (6) MONTHS from the mailing date of this communication.
- Failure to reply within the set or extended period for reply will, by statute, cause the application to become ABANDONED (35 U.S.C. § 133). Any reply received by the Office later than three months after the mailing date of this communication, even if timely filed, may reduce any earned patent term adjustment. See 37 CFR 1.704(b).

Status

1)☒ Responsive to communication(s) filed on *3-11-2009*.

2a)☒ This action is **FINAL**. 2b)☐ This action is non-final.

3)☐ Since this application is in condition for allowance except for formal matters, prosecution as to the merits is closed in accordance with the practice under *Ex parte Quayle*, 1935 C.D. 11, 453 O.G. 213.

Disposition of Claims

4)☒ Claim(s) *1-67* is/are pending in the application.

4a) Of the above claim(s) _____ is/are withdrawn from consideration.

5)☐ Claim(s) _____ is/are allowed.

6)☒ Claim(s) *1-67* is/are rejected.

7)☐ Claim(s) _____ is/are objected to.

8)☐ Claim(s) _____ are subject to restriction and/or election requirement.

Application Papers

9)☐ The specification is objected to by the Examiner.

10)☐ The drawing(s) filed on _____ is/are: a)☐ accepted or b)☐ objected to by the Examiner.

Applicant may not request that any objection to the drawing(s) be held in abeyance. See 37 CFR 1.85(a).

Replacement drawing sheet(s) including the correction is required if the drawing(s) is objected to. See 37 CFR 1.121(d).

11)☐ The oath or declaration is objected to by the Examiner. Note the attached Office Action or form PTO-152.

Priority under 35 U.S.C. § 119

12)☐ Acknowledgment is made of a claim for foreign priority under 35 U.S.C. § 119(a)-(d) or (f).

a)☐ All b)☐ Some * c)☐ None of:

1.☐ Certified copies of the priority documents have been received.

2.☐ Certified copies of the priority documents have been received in Application No. _____.

3.☐ Copies of the certified copies of the priority documents have been received in this National Stage application from the International Bureau (PCT Rule 17.2(a)).

* See the attached detailed Office action for a list of the certified copies not received.

Attachment(s)

1) ☐ Notice of References Cited (PTO-892)
2) ☐ Notice of Draftsperson's Patent Drawing Review (PTO-948)
3) ☒ Information Disclosure Statement(s) (PTO/SB/08) Paper No(s)/Mail Date *3-31-2009*.
4) ☐ Interview Summary (PTO-413) Paper No(s)/Mail Date. _____.
5) ☐ Notice of Informal Patent Application
6) ☐ Other: _____.

DETAILED ACTION

1. This action is responding to application amendments filed on 3-11-2009.

2. Claims **1 - 67** are pending. Claims **1, 21, 38, 53, 66, 67** are independent. This application was filed on 2-27-2004.

Response to Arguments

3. Applicant's arguments have been fully considered but were not persuasive.

3.1 Applicant argues that the referenced prior art does not disclose, *"a first arithmetic circuit comprising a first plurality of arithmetic structures feeding back high order bits" of a previously executed arithmetic instruction in the public- key cryptography application, generated by the first arithmetic circuit, to a second arithmetic circuit comprising a second plurality of arithmetic structures". (Remarks Page 2); and remarks conceming similar multi-step arithmetic processing. (Remarks Pages 2, 3, 6, 7)*

Applicant's claimed invention is a sequence of arithmetic process steps (multiplication, addition, subtraction) performed and designated at completion as a single arithmetic instruction. Chen prior art discloses multiple arithmetic circuits used to perform arithmetic processing steps. Chen prior art discloses the same arithmetic process steps (multiple, addition, subtraction). In addition, Chen prior art discloses the concept of feedback whereby a circuit using feedback is defined as: *"the transfer of part of the output of an active circuit or device back to the input"*. The concept and usage of feedback enables the processing of a sequence of arithmetic process steps as a single

arithmetic operation or instruction. In feedback, the output of one arithmetic process step is input to a next arithmetic process step in a sequence of arithmetic processing steps. The completion of the set of multiple processing steps results in the completion of a single arithmetic complex operation or instruction.

This claim limitation indicates a sequence of arithmetic process steps resulting in a completed arithmetic operation. The concept of feedback enables a sequence of arithmetic process steps such as multiplication and addition operations to be performed. The concept of feedback allows the output of a process step to be used as input for a next process step. The single arithmetic operation or instruction would be the completion of the set of arithmetic processing steps.

The Examiner is unclear why with the usage of feedback *(see Remarks Page 3)*, as used by Chen prior art in the operation of arithmetic processing steps resulting in the completion of a single arithmetic operation or instruction, the claimed invention is not disclosed. The cited prior art indicates each specific arithmetic process step such as multiplication, subtraction, and addition that is performed by the claimed invention. With the usage of feedback the precise arithmetic instruction consisting of the specified required sequence of arithmetic processing steps can be achieved.

3.2 Applicant argues the *"Bushan prior art". (Remarks Page 4)*

Bhushan prior art discloses the concept of feedback (see definition from above) equivalent to Chen prior art.

Bushan prior art discloses that if a current instruction requires the result from a

previous instruction. The result can be supplied to a next instruction. This disclosure indicates that the results of one instruction are saved and used as input or as an operand for a next instruction. This disclosure is equivalent to the feedback disclosure of Chen and satisfies the claimed invention limitation of using the output of an instruction as input to the next instruction. There is an additional option where the result may be bypassed based on an equality comparison. But, the result may also be used.

3.3 Applicant argues that the referenced prior art does not disclose, *"supplying a third number to the second arithmetic circuit and the first partial result being a representation of the high order bits summed with low order bits of a result of a first number multiplied by a second number and with the third number". (Remarks Page 8)*

The indicated passage is a sequence of arithmetic processing steps addressed in the above response to remarks. The third number is the feedback from a previous arithmetic processing step and is supplied to a next arithmetic processing step. This sequence discloses a third step, but, the number of processing can be three or more.

Summary paragraph from above repeated:

The Examiner is unclear why with the usage of feedback *(see Remarks Page 3)*, as used by Chen prior art in the operation of arithmetic processing steps resulting in the completion of a single arithmetic operation or instruction, the claimed invention is not disclosed. The cited prior art indicates each specific arithmetic process step such as multiplication, subtraction, and addition that is performed by the claimed invention. With the usage of feedback the precise arithmetic instruction consisting of the specified required sequence of arithmetic processing steps can be achieved.

Claim Rejections - 35 USC § 103

4. The following is a quotation of 35 U.S.C. 103(a) which forms the basis for all obviousness rejections set forth in this Office action:

> a) A patent may not be obtained though the invention is not identically disclosed or described as set forth in section 102 of this title, if the differences between the subject matter sought to be patented and the prior art are such that the subject matter as a whole would have been obvious at the time the invention was made to a person having ordinary skill in the art to which said subject matter pertains. Patentability shall not be negatived by the manner in which the invention was made.

5. Claims **1, 4 - 10, 19, 21 - 26, 36, 38 - 42, 48, 52 - 60, 62, 66, 67** are rejected under 35 U.S.C. 103(a) as being unpatentable over **Chen et al.** (US Patent No. **6,763,365**) in view of **Bhushan et al.** (US PGPUB No. **20020174157**).

With Regards to Claim 1, Chen discloses a method implemented in a device supporting a public-key cryptography application (see Chen col. 6, lines 23-25: arithmetic operations to support acceleration of cryptographic functions (cryptographic calculations)), the method comprising: a first arithmetic circuit comprising a first plurality of arithmetic structures feeding back high order bits of an executed arithmetic instruction in the public-key cryptography application, generated by the first arithmetic circuit, to a second arithmetic circuit comprising a second plurality of arithmetic structures (see Chen col. 11, lines 34-40: feedback; first using circuit; then using circuit again with register provided with output from first operational stage; col. 10, lines 13-26: multiple-accumulate instruction; first addend comes from the rightmost k bits of Z register; bits are added to the k bits in the rightmost portion of the product A,B); and the second

arithmetic circuit, generating a first partial result of a currently executing arithmetic instruction in the public-key cryptography application, the first partial result representing the high order bits summed with low order bits of a result of a first number multiplied by a second number, the summing of the high order bits being performed during multiplication of the first number and the second number, the summing and at least a portion of the multiplication being performed in the second arithmetic circuit; storing the first partial result; and using the stored first partial result in a subsequent computation in the public-key cryptography application. (see Chen col. 4, lines 8-11: multiplication and addition are performed by large circuits; col. 10, lines 13-36; col. 11, lines 34-40: feedback; first using circuit; then using circuit again with register provided with output from first operational stage (multiplication with feedback))

Bhushan discloses wherein a previously executed single arithmetic instruction. (see Bhushan paragraph [0116], lines 14-20: instruction requires an operand, a result from a previous instructions, the result may be bypassed under the direction of bypass routing control)

It would have been obvious to one of ordinary skill in the art to modify Chen for a previously executed single arithmetic instruction to generate a result as taught by Bhushan. One of ordinary skill in the art would have been motivated to employ the teachings of Bhushan in order to provide an efficient method for an uncomplicated arithmetic circuit that is capable of adding or subtracting numbers in redundant from and comparing a result without requiring propagation of carry signals. (see Bhushan paragraph [0062], lines 1-5)

With Regards to Claim 4, Chen discloses the method as recited in claim 1 further comprising feeding back the high order bits through a register to the second arithmetic circuit. (see Chen col. 11, lines 34-40: feedback; first using circuit; then using circuit again with register provided with output from first operational stage; col. 10, lines 13-26: multiple-accumulate instruction; first addend comes from the rightmost k bits of Z register; bits are added to the k bits in the rightmost portion of the product A,B)

With Regards to Claim 5, Chen discloses the method as recited in claim 1, further comprising: generating a second partial result of the currently executing arithmetic instruction in the first arithmetic circuit, the second partial result representing the high order bits of the multiplication result of the first number multiplied by the second number. (see Chen col. 11, lines 34-40: feedback; first using circuit; then using circuit again with register provided with output from first operational stage; col. 10, lines 13-26: multiple-accumulate instruction; first addend comes from the rightmost k bits of Z register; bits are added to the k bits in the rightmost portion of the product A,B)

With Regards to Claim 6, Chen discloses the method as recited in claim 1 further comprising: generating a second partial result of the currently executing arithmetic instruction, the second partial result representing the high order bits of the multiplication result of the first number multiplied by the second number summed with the high order bits of the executed arithmetic instruction. (see Chen col. 11, lines 34-40: feedback;

first using circuit; then using circuit again with register provided with output from first operational stage; col. 10, lines 13-26: multiple-accumulate instruction; first addend comes from the rightmost k bits of Z register; bits are added to the k bits in the rightmost portion of the product A,B)

Bhushan discloses wherein the previously executed arithmetic instruction. (see Bhushan paragraph [0116], lines 14-20: instruction requires an operand, a result from a previous instructions, the result may be bypassed under the direction of bypass routing control)

It would have been obvious to one of ordinary skill in the art to modify Chen for the previously executed arithmetic instruction as taught by Bhushan. One of ordinary skill in the art would have been motivated to employ the teachings of Bhushan in order to provide an efficient method for an uncomplicated arithmetic circuit that is capable of adding or subtracting numbers in redundant from and comparing a result without requiring propagation of carry signals. (see Bhushan paragraph [0062], lines 1-5)

With Regards to Claim 7, Chen discloses the method as recited in claim 6 further comprising supplying values generated in one or more most significant columns of the second arithmetic structures to one or more least significant columns of the first arithmetic structures while generating the first and second partial results. (see Chen col. 11, lines 34-40: feedback; first using circuit; then using circuit again with register provided with output from first operational stage; col. 10, lines 13-26: multiple-accumulate instruction; first addend comes from the rightmost k bits of Z register; bits

are added to the k bits in the rightmost portion of the product A,B)

With Regards to Claim 8, Chen discloses the method as recited in claim 5 wherein the generating of the first and second partial result is in response to execution of a single arithmetic instruction. (see Chen col. 11, lines 34-40: feedback; first using circuit; then using circuit again with register provided with output from first operational stage; col. 10, lines 13-26: multiple-accumulate instruction; first addend comes from the rightmost k bits of Z register; bits are added to the k bits in the rightmost portion of the product A,B)

With Regards to Claim 9, Chen discloses the method as recited in claim 6 wherein the generating of the first and second partial result is in response to execution of a single arithmetic instruction. (see Chen col. 11, lines 34-40: feedback; first using circuit; then using circuit again with register provided with output from first operational stage; col. 10, lines 13-26: multiple-accumulate instruction; first addend comes from the rightmost k bits of Z register; bits are added to the k bits in the rightmost portion of the product A,B)

With Regards to Claim 10, Chen discloses the method as recited in claim 1 wherein at least one of the first and second pluralities of arithmetic structures comprises a plurality of carry save adder tree columns. (see Chen (see Chen col. 4, lines 8-11: multiplication and addition are performed by large circuits; col. 14, lines 54-59: carry-out signal from adder; supplies a second carry-out signal to next processing element)

With Regards to Claim 19, Chen discloses the method as recited in claim 1, further comprising feeding back high order bits of the currently executing arithmetic instruction from the first arithmetic circuit to the second arithmetic circuit for use with execution of a subsequent single arithmetic instruction. (see Chen col. 11, lines 34-40: feedback; first using circuit; then using circuit again with register provided with output from first operational stage; col. 10, lines 13-26: multiple-accumulate instruction; first addend comes from the rightmost k bits of Z register; bits are added to the k bits in the rightmost portion of the product A,B)

With Regards to Claim 21, Chen discloses a method implemented in a device supporting public-key cryptography application (see Chen col. 6, lines 23-25: arithmetic operations to support acceleration of cryptographic functions), the method comprising: a first arithmetic circuit comprising a first plurality of arithmetic structures feeding back high order bits of an executed arithmetic instruction in the public-key cryptography application, generated by the first arithmetic circuit to a second arithmetic circuit comprising a second plurality of arithmetic structures; supplying a third number to the second arithmetic circuit; the second arithmetic circuit generating a first partial result of a currently executing arithmetic instruction in the public-key cryptography application, the first partial result being a representation of the high order bits summed with, low order bits of a result of a first number multiplied by a second number, and with the third number, the summing being performed during multiplication of the first number and the second number, the summing and at least a portion of the multiplication being

performed in the second arithmetic circuit (see Chen col. 11, lines 34-40: feedback; first using circuit; then using circuit again with register provided with output from first operational stage; col. 10, lines 13-26: multiple-accumulate instruction; first addend comes from the rightmost k bits of Z register; bits are added to the k bits in the rightmost portion of the product A,B); storing the first partial result; and using the first partial result in a subsequent computation in the public-key cryptography application. (see Chen col. 4, lines 8-11: multiplication and addition are performed by large circuits; col. 10, lines 13-36; col. 11, lines 34-40: feedback; first using circuit; then using circuit again with register provided with output from first operational stage (multiplication with feedback); col. 6, lines 12-14; distinct operands in pipelined mode (instructions); col. 10, lines 15-23: low order k bits from multiplier are supplied to adder)

Bhushan discloses wherein a previously executed arithmetic instruction. (see Bhushan paragraph [0116], lines 14-20: instruction requires an operand, a result from a previous instructions, the result may be bypassed under the direction of bypass routing control)

It would have been obvious to one of ordinary skill in the art to modify Chen for a previously executed arithmetic instruction as taught by Bhushan. One of ordinary skill in the art would have been motivated to employ the teachings of Bhushan in order to provide an efficient method for an uncomplicated arithmetic circuit that is capable of adding or subtracting numbers in redundant from and comparing a result without requiring propagation of carry signals. (see Bhushan paragraph [0062], lines 1-5)

With Regards to Claim 22, Chen discloses the method as recited in claim 21 further

comprising feeding back the high order bits through a register to the second arithmetic circuit. (see Chen col. 4, lines 8-11: multiplication and addition are performed by large circuits; col. 10, lines 13-36; col. 11, lines 34-40: feedback; first using circuit; then using circuit again with register provided with output from first operational stage (multiplication with feedback); col. 6, lines 23-25: arithmetic operations to support acceleration of cryptographic functions)

With Regards to Claim 23, Chen discloses the method as recited in claim 21, further comprising: the first arithmetic circuit generating a second partial result of the currently executing arithmetic instruction, the second partial result representing the high order bits of the multiplication result of the first number multiplied by the second number. (see Chen col. 4, lines 8-11: multiplication and addition are performed by large circuits; col. 10, lines 13-36; col. 11, lines 34-40: feedback; first using circuit; then using circuit again with register provided with output from first operational stage (multiplication with feedback); col. 6, lines 23-25: arithmetic operations to support acceleration of cryptographic functions)

With Regards to Claim 24, Chen discloses the method as recited in claim 21 further comprising: generating a second partial result of the currently executing arithmetic instruction, the second partial result representing the high order bits of the multiplication result of the first number multiplied by the second number summed with the high order bits of the executed arithmetic instruction and the third number. (see Chen col. 4, lines

8-11: multiplication and addition are performed by large circuits; col. 10, lines 13-36; col. 11, lines 34-40: feedback; first using circuit; then using circuit again with register provided with output from first operational stage (multiplication with feedback); col. 6, lines 23-25: arithmetic operations to support acceleration of cryptographic functions)

Bhushan discloses wherein the previously executed arithmetic instruction. (see Bhushan paragraph [0116], lines 14-20: instruction requires an operand, a result from a previous instructions, the result may be bypassed under the direction of bypass routing control)

It would have been obvious to one of ordinary skill in the art to modify Chen for the previously executed arithmetic instruction as taught by Bhushan. One of ordinary skill in the art would have been motivated to employ the teachings of Bhushan in order to provide an efficient method for an uncomplicated arithmetic circuit that is capable of adding or subtracting numbers in redundant from and comparing a result without requiring propagation of carry signals. (see Bhushan paragraph [0062], lines 1-5)

With Regards to Claim 25, Chen discloses the method as recited in claim 24 further comprising supplying values generated in one or more most significant columns of the second arithmetic structures to one or more least significant columns of the first arithmetic structures while generating the first and second partial results. (see Chen col. 4, lines 8-11: multiplication and addition are performed by large circuits; col. 10, lines 13-36; col. 11, lines 34-40: feedback; first using circuit; then using circuit again with register provided with output from first operational stage (multiplication with feedback);

col. 6, lines 23-25: arithmetic operations to support acceleration of cryptographic functions)

With Regards to Claim 26, Chen discloses the method as recited in claim 23 wherein the generating of the first and second partial result is in response to execution of a single arithmetic instruction. (see Chen col. 11, lines 34-40: feedback; first using circuit; then using circuit again with register provided with output from first operational stage; col. 10, lines 13-26: multiple-accumulate instruction; first addend comes from the rightmost k bits of Z register; bits are added to the k bits in the rightmost portion of the product A,B)

With Regards to Claim 36, Chen discloses the method as recited in claim 21 further comprising feeding back high order bits of the currently executing arithmetic instruction from the first arithmetic circuit to the second arithmetic circuit for use with execution of a subsequent single arithmetic instruction. (see Chen col. 11, lines 34-40: feedback; first using circuit; then using circuit again with register provided with output from first operational stage; col. 10, lines 13-26: multiple-accumulate instruction; first addend comes from the rightmost k bits of Z register; bits are added to the k bits in the rightmost portion of the product A,B)

With Regards to Claim 38, Chen discloses a processor configured to support public-key cryptography applications (see Chen col. 6, lines 23-25: arithmetic operations to

support acceleration of cryptographic functions), comprising: a first plurality of arithmetic structures configured to generate high order bits for an arithmetic operation in a public-key cryptography application that includes a multiplication operation; and a second plurality of arithmetic structures configured to generate low order bits of the arithmetic operation; wherein the second arithmetic structures are further configured to receive the high order bits generated by the first plurality of arithmetic structures during a previous arithmetic operation in the public-key cryptography application and to generate a first partial result of the arithmetic operation, the first partial result representing the high order bits summed with low order bits of a multiplication result of the multiplication operation; and wherein the processor further comprises a register configured to store the first partial result for use in a subsequent arithmetic operation in the public-key cryptography application. (see Chen col. 4, lines 8-11: multiplication and addition are performed by large circuits; col. 10, lines 13-36; col. 11, lines 34-40: feedback; first using circuit; then using circuit again with register provided with output from first operational stage (multiplication with feedback))

With Regards to Claim 39, Chen discloses the processor as recited in claim 38, wherein the first arithmetic structures are configured to generate a second partial result of the arithmetic instruction, the second partial result representing the high order bits of the arithmetic operation. (see Chen col. 11, lines 34-40: feedback; first using circuit; then using circuit again with register provided with output from first operational stage; col. 10, lines 13-26: multiple-accumulate instruction; first addend comes from the

rightmost k bits of Z register; bits are added to the k bits in the rightmost portion of the product A,B)

With Regards to Claim 40, Chen discloses the processor as recited in claim 39, wherein the second arithmetic structures are further configured to supply values generated in one or more most significant columns of the second arithmetic structures to one or more least significant columns of the first arithmetic structures while generating the first and second partial results. (see Chen col. 6, lines 12-14; distinct operands in pipelined mode (instructions); col. 10, lines 15-23: low order k bits from multiplier are supplied to adder)

With Regards to Claim 41, Chen discloses the processor as recited in claim 39, wherein the first and second arithmetic structures are configured to generate of the first and second partial results in response to execution of a single arithmetic instruction. (see Chen col. 11, lines 34-40: feedback; first using circuit; then using circuit again with register provided with output from first operational stage; col. 10, lines 13-26: multiple-accumulate instruction; first addend comes from the rightmost k bits of Z register; bits are added to the k bits in the rightmost portion of the product A,B)

With Regards to Claim 42, Chen discloses the processor as recited in claim 38, further comprising a register coupled to the first and second arithmetic structures to supply the

high order bits to the second arithmetic structures. (see Chen col. 11, lines 34-40: feedback; first using circuit; then using circuit again with register provided with output from first operational stage; col. 10, lines 13-26: multiple-accumulate instruction; first addend comes from the rightmost k bits of Z register; bits are added to the k bits in the rightmost portion of the product A,B)

With Regards to Claim 48, Chen discloses the processor as recited in claim 38, wherein at least one of the first and second pluralities of arithmetic structures comprises a plurality of carry save adder tree columns. (see Chen col. 14, lines 54-59: carry-out signal from adder; supplies a second carry-out signal to next processing element)

With Regards to Claim 52, Chen discloses the processor as recited in claim 38, wherein the processor is a general purpose processor. (see Chen col. 20, lines 32-34: generic hardware processor element)

With Regards to Claim 53, Chen discloses a processor configured to support public-key cryptography applications (see Chen col. 6, lines 23-25: arithmetic operations to support acceleration of cryptographic functions), comprising: a first plurality of arithmetic structures configured to generate high order bits for an arithmetic operation in a public-key cryptography application that includes a multiplication operation of a first and a second number (see Chen col. 11, lines 34-40: feedback; first using circuit; then using circuit again with register provided with output from first operational stage; col. 10, lines

13-26: multiple-accumulate instruction; first addend comes from the rightmost k bits of Z register; bits are added to the k bits in the rightmost portion of the product A,B); a second plurality of arithmetic structures configured to generate low order bits of the arithmetic operation; wherein the second arithmetic structures are configured to: receive the high order bits generated by the first plurality of arithmetic structures during a previous arithmetic operation; receive a third number; and generate a first partial result of the arithmetic operation, the first partial result representing the high order bits summed with low order bits of a multiplication result of the multiplication operation, and with the third number; and wherein the processor further comprises a register configured to store the first partial result for use in a subsequent arithmetic operation in the public-key cryptography application. (see Chen col. 4, lines 8-11: multiplication and addition are performed by large circuits; col. 10, lines 13-36; col. 11, lines 34-40: feedback; first using circuit; then using circuit again with register provided with output from first operational stage (multiplication with feedback))

With Regards to Claim 54, Chen discloses the processor as recited in claim 53, wherein the first arithmetic structures are further configured to generate a second partial result of the arithmetic instruction, the second partial result representing the high order bits of the arithmetic operation. (see Chen col. 11, lines 34-40: feedback; first using circuit; then using circuit again with register provided with output from first operational stage; col. 10, lines 13-26: multiple-accumulate instruction; first addend comes from the rightmost k bits of Z register; bits are added to the k bits in the rightmost portion of the

product A,B)

With Regards to Claim 55, Chen discloses the processor as recited in claim 54, wherein the second arithmetic structures are further configured to generate values in one or more most significant columns and to supply them to one or more least significant columns of the first arithmetic structures while generating the first and second partial results. (see Chen col. 11, lines 34-40: feedback; first using circuit; then using circuit again with register provided with output from first operational stage; col. 10, lines 13-26: multiple-accumulate instruction; first addend comes from the rightmost k bits of Z register; bits are added to the k bits in the rightmost portion of the product A,B)

With Regards to Claim 56, Chen discloses the processor as recited in claim 54, wherein the first arithmetic structures are configured to generate of the first and second partial result in response to execution of a single arithmetic instruction. (see Chen col. 11, lines 34-40: feedback; first using circuit; then using circuit again with register provided with output from first operational stage; col. 10, lines 13-26: multiple-accumulate instruction; first addend comes from the rightmost k bits of Z register; bits are added to the k bits in the rightmost portion of the product A,B)

With Regards to Claim 57, Chen discloses the processor as recited in claim 53, further comprising a register coupled to the first and second arithmetic structures to supply the high order bits to the second arithmetic structures. (see Chen col. 11, lines 34-40:

feedback; first using circuit; then using circuit again with register provided with output from first operational stage; col. 10, lines 13-26: multiple-accumulate instruction; first addend comes from the rightmost k bits of Z register; bits are added to the k bits in the rightmost portion of the product A,B)

With Regards to Claim 58, Chen discloses the processor as recited in claim 53, further comprising an adder circuit configured to receive the first partial result and to generate a non redundant representation of the first partial result and a carry out value. (see Chen col. 11, lines 34-40: feedback; first using circuit; then using circuit again with register provided with output from first operational stage; col. 10, lines 13-26: multiple-accumulate instruction; first addend comes from the rightmost k bits of Z register; bits are added to the k bits in the rightmost portion of the product A,B; col. 14, lines 54-59: carry-out signal from adder; supplies a second carry-out signal to next processing element)

With Regards to Claim 59, Chen discloses the processor as recited in claim 58, wherein the adder circuit is further configured to feed the carry out value back to itself as an input. (see Chen col. 14, lines 54-59: carry-out signal from adder; supplies a second carry-out signal to next processing element)

With Regards to Claim 60, Chen discloses the processor as recited in claim 58, method wherein the adder circuit is further configured to feed the carry out value back to

the second arithmetic structures. (see Chen col. 14, lines 54-59: carry-out signal from adder; supplies a second carry-out signal to next processing element)

With Regards to Claim 62, Chen discloses the processor as recited in claim 53, wherein at least one of the first and second arithmetic structures comprises carry save adder tree columns. (see Chen col. 14, lines 54-59: carry-out signal from adder; supplies a second carry-out signal to next processing element)

With Regards to Claim 66, Chen discloses an apparatus configured to support a public-key cryptography application (see Chen col. 6, lines 23-25: arithmetic operations to support acceleration of cryptographic functions) comprising: means for feeding back high order bits of an executed arithmetic instruction, generated by a first arithmetic circuit, to a second arithmetic circuit generating low order bits of a currently executing arithmetic instruction; means for using the second arithmetic circuit to generate a first partial result of the currently executing arithmetic instruction, the first partial result representing the high order bits of the executed arithmetic instruction that are summed with low order bits of a multiplication result of a first number multiplied by a second number; means for using the first partial result in a subsequent computation in the public-key cryptography application. (see Chen col. 4, lines 8-11: multiplication and addition are performed by large circuits; col. 10, lines 13-36; col. 11, lines 34-40: feedback; first using circuit; then using circuit again with register provided with output from first operational stage (multiplication with feedback); col. 6, lines 23-25: arithmetic

operations to support acceleration of cryptographic functions)

Bhushan discloses wherein a previously executed arithmetic instruction. (see Bhushan paragraph [0116], lines 14-20: instruction requires an operand, a result from a previous instructions, the result may be bypassed under the direction of bypass routing control)

It would have been obvious to one of ordinary skill in the art to modify Chen for a previously executed arithmetic instruction as taught by Bhushan. One of ordinary skill in the art would have been motivated to employ the teachings of Bhushan in order to provide an efficient method for an uncomplicated arithmetic circuit that is capable of adding or subtracting numbers in redundant from and comparing a result without requiring propagation of carry signals. (see Bhushan paragraph [0062], lines 1-5)

With Regards to Claim 67, Chen discloses an apparatus configured to support a public-key cryptography application (see Chen col. 6, lines 23-25: arithmetic operations to support acceleration of cryptographic functions) comprising: means for feeding back high order bits of an executed arithmetic instruction, from a first arithmetic circuit that generated the high order bits (see Chen col. 11, lines 34-40: feedback; first using circuit; then using circuit again with register provided with output from first operational stage; col. 10, lines 13-26: multiple-accumulate instruction; first addend comes from the rightmost k bits of Z register; bits are added to the k bits in the rightmost portion of the product A,B), to a second arithmetic circuit generating low order bits of a currently executing arithmetic instruction (see Chen col. 6, lines 12-14; distinct operands in pipelined mode (instructions); col. 10, lines 15-23: low order k bits from multiplier are

supplied to adder); means for supplying a third number to the second arithmetic circuit; and means for using the second arithmetic circuit to generate a first partial result, the first partial result being a representation of the high order bits of the executed arithmetic instruction summed with low order bits of a result of a first number multiplied by a second number and with the third number; and means for using the first partial result in a subsequent computation in the public-key cryptography application. (see Chen col. 4, lines 8-11: multiplication and addition are performed by large circuits; col. 10, lines 13-36; col. 11, lines 34-40: feedback; first using circuit; then using circuit again with register provided with output from first operational stage (multiplication with feedback); col. 6, lines 23-25: arithmetic operations to support acceleration of cryptographic functions)

Bhushan discloses wherein the previously executed single arithmetic instruction. (see Bhushan paragraph [0116], lines 14-20: instruction requires an operand, a result from a previous instructions, the result may be bypassed under the direction of bypass routing control)

It would have been obvious to one of ordinary skill in the art to modify Chen for the previously executed single arithmetic instruction as taught by Bhushan. One of ordinary skill in the art would have been motivated to employ the teachings of Bhushan in order to provide an efficient method for an uncomplicated arithmetic circuit that is capable of adding or subtracting numbers in redundant from and comparing a result without requiring propagation of carry signals. (see Bhushan paragraph [0062], lines 1-5)

6. Claims **2, 3, 15 - 18, 27 - 29, 35, 43 - 46** are rejected under 35 U.S.C. 103(a) as being unpatentable over **Chen-Bhushan** and further in view of **Lasher et al.** (US Patent No. **4,863,247**).

With Regards to Claim 2, Chen discloses the method as recited in claim 1 wherein the high order bits are fed back. (see Chen col. 11, lines 34-40: feedback; first using circuit; then using circuit again with register provided with output from first operational stage; col. 10, lines 13-26: multiple-accumulate instruction; first addend comes from the rightmost k bits of Z register; bits are added to the k bits in the rightmost portion of the product A,B) And, Lasher discloses wherein the result is in redundant number representation. (see Lasher col. 6, lines 6-9; col. 6, lines 16-18: redundant number representations)

It would have been obvious to one of ordinary skill in the art to modify Chen for a result in redundant number representation as taught by Lasher. One of ordinary skill in the art would have been motivated to employ the teachings of Lasher in order that fully parallel carry-free operation is provided for with reduced complexity. (see Lasher col. 2, lines 57-62)

With Regards to Claim 3, Chen discloses the method as recited in claim 2 wherein the redundant number representation includes sum and carry bits. (see Chen col. 14, lines 54-59: carry-out signal from adder; supplies a second carry-out signal to next processing element) And, Lasher discloses wherein the result is in redundant number

representation. (see Lasher col. 6, lines 6-9; col. 6, lines 16-18: redundant number representations)

It would have been obvious to one of ordinary skill in the art to modify Chen for a result in redundant number representation as taught by Lasher. One of ordinary skill in the art would have been motivated to employ the teachings of Lasher in order that fully parallel carry-free operation is provided for with reduced complexity. (see Lasher col. 2, lines 57-62)

With Regards to Claim 15, Chen discloses the method as recited in claim 1 wherein the first partial result. (see Chen col. 11, lines 34-40: feedback; first using circuit; then using circuit again with register provided with output from first operational stage; col. 10, lines 13-26: multiple-accumulate instruction; first addend comes from the rightmost k bits of Z register; bits are added to the k bits in the rightmost portion of the product A,B) And, Lasher discloses wherein the result is in redundant number representation. (see Lasher col. 6, lines 6-9; col. 6, lines 16-18: redundant number representations)

It would have been obvious to one of ordinary skill in the art to modify Chen for a result in redundant number representation as taught by Lasher. One of ordinary skill in the art would have been motivated to employ the teachings of Lasher in order that fully parallel carry-free operation is provided for with reduced complexity. (see Lasher col. 2, lines 57-62)

With Regards to Claim 16, Chen discloses the method as recited in claim 15 further

comprising supplying the first partial result to an adder circuit to generate the first partial result and a carry out value. (see Chen (see Chen col. 11, lines 34-40: feedback; first using circuit; then using circuit again with register provided with output from first operational stage; col. 10, lines 13-26: multiple-accumulate instruction; first addend comes from the rightmost k bits of Z register; bits are added to the k bits in the rightmost portion of the product A,B; col. 14, lines 54-59: carry-out signal from adder; supplies a second carry-out signal to next processing element) And, Lasher discloses wherein the result is a non redundant representation. (see Lasher col. 6, lines 6-9; col. 6, lines 16-18: redundant number representations)

It would have been obvious to one of ordinary skill in the art to modify Chen for a result in redundant number representation as taught by Lasher. One of ordinary skill in the art would have been motivated to employ the teachings of Lasher in order that fully parallel carry-free operation is provided for with reduced complexity. (see Lasher col. 2, lines 57-62)

With Regards to Claim 17, Chen discloses the method as recited in claim 16 further comprising feeding back the carry out value to the adder circuit. (see Chen col. 14, lines 54-59: carry-out signal from adder; supplies a second carry-out signal to next processing element)

With Regards to Claim 18, Chen discloses the method as recited in claim 16, further comprising feeding back the carry out value to the second arithmetic circuit. (see Chen

col. 14, lines 54-59: carry-out signal from adder; supplies a second carry-out signal to next processing element)

With Regards to Claim 27, Chen discloses the method as recited in claim 21 supplying the first partial result to an adder circuit to generate a non redundant representation of the first partial result and a carry out value. (see Chen col. 14, lines 54-59: carry-out signal from adder; supplies a second carry-out signal to next processing element) And, Lasher discloses wherein the result is a non redundant number representation. (see Lasher col. 6, lines 6-9; col. 6, lines 16-18: redundant number representations)

It would have been obvious to one of ordinary skill in the art to modify Chen for a result is a non redundant number representation as taught by Lasher. One of ordinary skill in the art would have been motivated to employ the teachings of Lasher in order that fully parallel carry-free operation is provided for with reduced complexity. (see Lasher col. 2, lines 57-62)

With Regards to Claim 28, Chen discloses the method as recited in claim 27 further comprising feeding back the carry out value to the adder circuit. (see Chen col. 14, lines 54-59: carry-out signal from adder; supplies a second carry-out signal to next processing element)

With Regards to Claim 29, Chen discloses the method as recited in claim 27, method further comprising feeding back the carry out value to the second arithmetic structures.

(see Chen col. 14, lines 54-59: carry-out signal from adder; supplies a second carry-out signal to next processing element)

With Regards to Claim 35, Chen discloses the method as recited in claim 21 wherein the high order bits. (see Chen col. 11, lines 34-40: feedback; first using circuit; then using circuit again with register provided with output from first operational stage; col. 10, lines 13-26: multiple-accumulate instruction; first addend comes from the rightmost k bits of Z register; bits are added to the k bits in the rightmost portion of the product A,B) And, Lasher discloses wherein the result is in redundant number representation. (see Lasher col. 6, lines 6-9; col. 6, lines 16-18: redundant number representations)

It would have been obvious to one of ordinary skill in the art to modify Chen for a result in redundant number representation as taught by Lasher. One of ordinary skill in the art would have been motivated to employ the teachings of Lasher in order that fully parallel carry-free operation is provided for with reduced complexity. (see Lasher col. 2, lines 57-62)

With Regards to Claim 43, Chen discloses the processor as recited in claim 38, wherein the first partial result. (see Chen col. 11, lines 34-40: feedback; first using circuit; then using circuit again with register provided with output from first operational stage; col. 10, lines 13-26: multiple-accumulate instruction; first addend comes from the rightmost k bits of Z register; bits are added to the k bits in the rightmost portion of the product A,B) And, Lasher discloses wherein the result is in redundant number

representation. (see Lasher col. 6, lines 6-9; col. 6, lines 16-18: redundant number representations)

It would have been obvious to one of ordinary skill in the art to modify Chen for a result in redundant number representation as taught by Lasher. One of ordinary skill in the art would have been motivated to employ the teachings of Lasher in order that fully parallel carry-free operation is provided for with reduced complexity. (see Lasher col. 2, lines 57-62)

With Regards to Claim 44, Chen discloses the processor as recited in claim 43, further comprising an adder circuit configured to receive the first partial result and to generate a non redundant representation of the first partial result and a carry out value. (see Chen col. 14, lines 54-59: carry-out signal from adder; supplies a second carry-out signal to next processing element)

With Regards to Claim 45, Chen discloses the processor as recited in claim 44, wherein adder circuit is configured to feed the carry out value back to itself as an input. (see Chen col. 14, lines 54-59: carry-out signal from adder; supplies a second carry-out signal to next processing element)

With Regards to Claim 46, Chen discloses the processor as recited in claim 44, wherein adder circuit is configured to feed the carry out value back to the second arithmetic structures. (see Chen col. 14, lines 54-59: carry-out signal from adder;

supplies a second carry-out signal to next processing element)

7. Claims **11, 20, 30, 31, 37, 47, 61** are rejected under 35 U.S.C. 103(a) as being unpatentable over **Chen-Bhushan** and further in view of **Stribaek et al.** (US Patent No. **7,181,484**).

With Regards to Claim 11, Chen discloses the method as recited in claim 1 wherein at least one of the first and second pluralities of arithmetic structures. (see Chen col. 4, lines 8-11: multiplication and addition are performed by large circuits; col. 10, lines 13-36; col. 11, lines 34-40: feedback; first using circuit; then using circuit again with register provided with output from first operational stage (multiplication with feedback); col. 6, lines 23-25: arithmetic operations to support acceleration of cryptographic functions) Chen does not specifically disclose whereby a plurality of Wallace tree columns. However, Stribaek discloses wherein further comprises a plurality of Wallace tree columns. (see Stribaek col. 9, lines 10-24; col. 9, lines 37-39: Wallace tree)

It would have been obvious to one of ordinary skill in the art to modify Chen for usage of Wallace tree multiplication as taught by Stribaek. One of ordinary skill in the art would have been motivated to employ the teachings of Stribaek in order to enable the capability for extended precision in arithmetic calculations due to extensive and increasing usage of public key cryptography. (see Stribaek col. 1, lines 61-67: " ... *Public-key cryptosystems have been used extensively for user authentication and secure key exchange, while private-key cryptography has been used extensively to*

encrypt communication channels. As the use of public-key cryptosystems increases, it becomes desirable to increase the performance of extended-precision modular arithmetic calculations. ... ")

With Regards to Claim 20, Chen discloses the method as recited claim 1 further comprising storing the high order bits. (see Chen col. 11, lines 34-40: feedback; first using circuit; then using circuit again with register provided with output from first operational stage; col. 10, lines 13-26: multiple-accumulate instruction; first addend comes from the rightmost k bits of Z register; bits are added to the k bits in the rightmost portion of the product A,B) Chen does not specifically disclose whereby an extended carry register. However, Stribaek discloses wherein an extended carry register. (see Stribaek col. 9, lines 10-24; col. 9, lines 37-39: Wallace tree; col. 5, lines 41-45: extended carry operations)

It would have been obvious to one of ordinary skill in the art to modify Chen for usage of extended carryoperations as taught by Stribaek. One of ordinary skill in the art would have been motivated to employ the teachings of Stribaek in order to enable the capability for extended precision in arithmetic calculations due to extensive and increasing usage of public key cryptography. (see Stribaek col. 1, lines 61-67)

With Regards to Claim 30, Chen discloses the method as recited in claim 21, wherein at least one of the first and second pluralities of arithmetic structures. (see Chen col. 4, lines 8-11: multiplication and addition are performed by large circuits; col. 10, lines 13-

36; col. 11, lines 34-40: feedback; first using circuit; then using circuit again with register provided with output from first operational stage (multiplication with feedback)) Chen does not specifically disclose a plurality of Wallace tree columns. However, Stribaek discloses wherein further comprises a plurality of Wallace tree columns. (see Stribaek col. 9, lines 10-24; col. 9, lines 37-39: Wallace tree; col. 2, line 66 - col. 3, line 6: public key cryptographic calculations)

It would have been obvious to one of ordinary skill in the art to modify Chen for usage of Wallace tree multiplication as taught by Stribaek. One of ordinary skill in the art would have been motivated to employ the teachings of Stribaek in order to enable the capability for extended precision in arithmetic calculations due to extensive and increasing usage of public key cryptography. (see Stribaek col. 1, lines 61-67)

With Regards to Claim 31, Chen discloses the method as recited in claim 21, wherein at least one of the first and second pluralities of arithmetic structures. (see Chen col. 4, lines 8-11: multiplication and addition are performed by large circuits; col. 10, lines 13-36; col. 11, lines 34-40: feedback; first using circuit; then using circuit again with register provided with output from first operational stage (multiplication with feedback); col. 14, lines 54-59: carry-out signal from adder; supplies a second carry-out signal to next processing element) Chen does not specifically disclose carry save adder tree columns. However, Stribaek discloses wherein further comprises a plurality of adder tree columns. (see Stribaek col. 9, lines 10-24; col. 9, lines 37-39: Wallace tree; col. 5, lines 41-45: extended carry operations; col. 7, lines 31-37; col. 9, lines 10-14: carry-

save adder; col. 2, line 66 - col. 3, line 6: public key cryptographic calculations)

It would have been obvious to one of ordinary skill in the art to modify Chen for usage of Wallace tree multiplication as taught by Stribaek. One of ordinary skill in the art would have been motivated to employ the teachings of Stribaek in order to enable the capability for extended precision in arithmetic calculations due to extensive and increasing usage of public key cryptography. (see Stribaek col. 1, lines 61-67)

With Regards to Claim 37, Chen discloses the method as recited in claim 21 further comprising storing the high order bits. (see Chen col. 11, lines 34-40: feedback; first using circuit; then using circuit again with register provided with output from first operational stage; col. 10, lines 13-26: multiple-accumulate instruction; first addend comes from the rightmost k bits of Z register; bits are added to the k bits in the rightmost portion of the product A,B) Chen does not specifically disclose whereby an extended carry register. However, Stribaek discloses wherein an extended carry register. (see Stribaek col. 9, lines 10-24; col. 9, lines 37-39: Wallace tree; col. 5, lines 41-45: extended carry operations)

It would have been obvious to one of ordinary skill in the art to modify Chen for usage of extended carry operations (register) as taught by Stribaek. One of ordinary skill in the art would have been motivated to employ the teachings of Stribaek in order to enable the capability for extended precision in arithmetic calculations due to extensive and increasing usage of public key cryptography. (see Stribaek col. 1, lines 61-67)

With Regards to Claim 47, Chen discloses the processor as recited in claim 38, wherein at least one of the first and second pluralities of the arithmetic structures. (see Chen col. 4, lines 8-11: multiplication and addition are performed by large circuits; col. 10, lines 13-36; col. 11, lines 34-40: feedback; first using circuit; then using circuit again with register provided with output from first operational stage (multiplication with feedback)) Chen does not specifically disclose whereby a plurality of Wallace tree columns. However, Stribaek discloses wherein further comprises a plurality of Wallace tree columns. (see Stribaek col. 9, lines 10-24; col. 9, lines 37-39: Wallace tree)

It would have been obvious to one of ordinary skill in the art to modify Chen for usage of Wallace tree multiplication as taught by Stribaek. One of ordinary skill in the art would have been motivated to employ the teachings of Stribaek in order to enable the capability for extended precision in arithmetic calculations due to extensive and increasing usage of public key cryptography. (see Stribaek col. 1, lines 61-67)

With Regards to Claim 61, Chen discloses the processor as recited in claim 53, wherein at least one of the first and second arithmetic structures. (see Chen col. 4, lines 8-11: multiplication and addition are performed by large circuits; col. 10, lines 13-36; col. 11, lines 34-40: feedback; first using circuit; then using circuit again with register provided with output from first operational stage (multiplication with feedback)) Chen does not specifically disclose whereby further comprising Wallace tree columns. However, Stribaek discloses wherein further comprises a Wallace tree column. (see Stribaek col. 9, lines 10-24; col. 9, lines 37-39: Wallace tree)

It would have been obvious to one of ordinary skill in the art to modify Chen as taught by Stribaek for usage of Wallace tree multiplication. One of ordinary skill in the art would have been motivated to employ the teachings of Stribaek in order to enable the capability for extended precision in arithmetic calculations due to extensive and increasing usage of public key cryptography. (see Stribaek col. 1, lines 61-67)

8. Claims **12 - 14, 32 - 34, 49 - 51, 63 - 65** are rejected under 35 U.S.C. 103(a) as being unpatentable over **Chen-Bhushan** and further in view of **Chen et al.** (US Patent No. **6,687,725**: referred to as "Chen2").

With Regards to Claim 12, Chen discloses the method as recited in claim 1 wherein at least one of the first and second pluralities of arithmetic structures is usable to perform integer multiplication. (see Chen col. 11, lines 34-40: feedback; first using circuit; then using circuit again with register provided with output from first operational stage; col. 10, lines 13-26: multiple-accumulate instruction; first addend comes from the rightmost k bits of Z register; bits are added to the k bits in the rightmost portion of the product A,B) And, Chen2 discloses wherein to perform XOR multiplication. (see Chen2 col. 4, line 64 - col. 5, line 2; col. 15, lines 29-31: XOR operations)

It would have been obvious to one of ordinary skill in the art to modify Chen to perform XOR multiplication as taught by Chen2. One of ordinary skill in the art would have been motivated to employ the teachings of Chen2 in order to provide an arithmetic circuit which can perform all arithmetic operations in the finite field, including addition,

multiplication, division, exponentiation and inverse multiplication. (see Chen2 col. 3, lines 17-21: "*... To solve the above mentioned problems, it is an object of the present invention to provide an arithmetic circuit which can perform all arithmetic operations in the finite field, including addition, multiplication, division, exponentiation and inverse multiplication. ...* ")

With Regards to Claim 13, Chen discloses the method as recited in claim 12, further comprising a logical circuit in at least one of the first and second arithmetic circuits supplying a variable value for integer multiplication mode that varies according to inputs supplied to the logical circuit if in integer multiplication mode, to thereby ensure a result unaffected by carry logic performing carries in integer multiplication mode. (see Chen col. 4, lines 8-11: multiplication and addition are performed by large circuits; col. 10, lines 13-36; col. 11, lines 34-40: feedback; first using circuit; then using circuit again with register provided with output from first operational stage (multiplication with feedback); col. 14, lines 54-59: carry-out signal from adder; supplies a second carry-out signal to next processing element) And, Chen2 discloses wherein supplying a fixed value if in XOR multiplication mode and to thereby ensure a result is determined in XOR multiplication. (see Chen2 col. 4, line 64 - col. 5, line 2; col. 15, lines 29-31: XOR operations)

It would have been obvious to one of ordinary skill in the art to modify Chen to support XOR operations as taught by Chen2. One of ordinary skill in the art would have been motivated to employ the teachings of Chen2 in order to provide an arithmetic

circuit which can perform all arithmetic operations in the finite field, including addition, multiplication, division, exponentiation and inverse multiplication. (see Chen2 col. 3, lines 17-21)

With Regards to Claim 14, Chen discloses the method as recited in claim 13 wherein the logical circuit operates as a majority circuit in integer multiplication mode. (see Chen col. 4, lines 8-11: multiplication and addition are performed by large circuits; col. 10, lines 13-36; col. 11, lines 34-40: feedback; first using circuit; then using circuit again with register provided with output from first operational stage (multiplication with feedback)) And, Chen2 discloses wherein outputs a zero in the XOR multiplication mode. (see Chen2 col. 4, line 64 - col. 5, line 2; col. 15, lines 29-31: XOR operations)

It would have been obvious to one of ordinary skill in the art to modify Chen to support XOR operations as taught by Chen2. One of ordinary skill in the art would have been motivated to employ the teachings of Chen2 in order to provide an arithmetic circuit which can perform all arithmetic operations in the finite field, including addition, multiplication, division, exponentiation and inverse multiplication. (see Chen2 col. 3, lines 17-21)

With Regards to Claim 32, Chen discloses the method as recited in claim 21, wherein at least one of the first and second pluralities of arithmetic structures is usable to perform integer multiplication. (see Chen col. 4, lines 8-11: multiplication and addition are performed by large circuits; col. 10, lines 13-36; col. 11, lines 34-40: feedback; first

using circuit; then using circuit again with register provided with output from first operational stage (multiplication with feedback); col. 6, lines 23-25: arithmetic operations to support acceleration of cryptographic functions) And, Chen2 discloses wherein perform both integer and XOR multiplication. (see Chen2 col. 4, line 64 - col. 5, line 2; col. 15, lines 29-31: XOR operations)

It would have been obvious to one of ordinary skill in the art to modify Chen to perform XOR multiplication as taught by Chen2. One of ordinary skill in the art would have been motivated to employ the teachings of Chen2 in order to provide an arithmetic circuit which can perform all arithmetic operations in the finite field, including addition, multiplication, division, exponentiation and inverse multiplication. (see Chen2 col. 3, lines 17-21)

With Regards to Claim 33, Chen discloses the method as recited in claim 32 further comprising a logic circuit in at least one of the first and second pluralities of arithmetic structures supplying a variable value that varies according to inputs supplied to the logical circuit if in integer multiplication mode, to thereby ensure a result unaffected by carry logic performing carries in integer multiplication mode. (see Chen col. 4, lines 8-11: multiplication and addition are performed by large circuits; col. 10, lines 13-36; col. 11, lines 34-40: feedback; first using circuit; then using circuit again with register provided with output from first operational stage (multiplication with feedback); col. 6, lines 23-25: arithmetic operations to support acceleration of cryptographic functions) And, Chen2 discloses wherein supplying a fixed value if in XOR multiplication mode.

(see Chen2 col. 4, line 64 - col. 5, line 2; col. 15, lines 29-31: XOR operations)

It would have been obvious to one of ordinary skill in the art to modify Chen to support XOR operations as taught by Chen2. One of ordinary skill in the art would have been motivated to employ the teachings of Chen2 in order to provide an arithmetic circuit which can perform all arithmetic operations in the finite field, including addition, multiplication, division, exponentiation and inverse multiplication. (see Chen2 col. 3, lines 17-21)

With Regards to Claim 34, Chen discloses the method as recited in claim 33 wherein the logic circuit operates as a majority circuit in integer multiplication mode. (see Chen col. 4, lines 8-11: multiplication and addition are performed by large circuits; col. 10, lines 13-36; col. 11, lines 34-40: feedback; first using circuit; then using circuit again with register provided with output from first operational stage (multiplication with feedback)) And, Chen2 discloses wherein outputs a zero in the XOR multiplication mode. (see Chen2 col. 4, line 64 - col. 5, line 2; col. 15, lines 29-31: XOR operations)

It would have been obvious to one of ordinary skill in the art to modify Chen to support XOR operations as taught by Chen2. One of ordinary skill in the art would have been motivated to employ the teachings of Chen2 in order to provide an arithmetic circuit which can perform all arithmetic operations in the finite field, including addition, multiplication, division, exponentiation and inverse multiplication. (see Chen2 col. 3, lines 17-21)

With Regards to Claim 49, Chen discloses the processor as recited in claim 38, wherein at least one of the first and second pluralities of arithmetic structures is configured to selectively perform one of integer multiplication according to a control signal. (see Chen col. 11, lines 34-40: feedback; first using circuit; then using circuit again with register provided with output from first operational stage; col. 10, lines 13-26: multiple-accumulate instruction; first addend comes from the rightmost k bits of Z register; bits are added to the k bits in the rightmost portion of the product A,B) And, Chen2 discloses wherein perform one of integer and XOR multiplication. (see Chen2 col. 4, line 64 - col. 5, line 2; col. 15, lines 29-31: XOR operations)

It would have been obvious to one of ordinary skill in the art to modify Chen to support XOR operations as taught by Chen2. One of ordinary skill in the art would have been motivated to employ the teachings of Chen2 in order to provide an arithmetic circuit which can perform all arithmetic operations in the finite field, including addition, multiplication, division, exponentiation and inverse multiplication. (see Chen2 col. 3, lines 17-21)

With Regards to Claim 50, Chen discloses the processor as recited in claim 49, further comprising a plurality of logic circuits in the first and second pluralities of arithmetic structures, each logic circuit responsive to the control signal to supply a variable output value in integer multiplication mode, the variable output value varying according to values of inputs supplied to the logic circuit, to thereby ensure a result unaffected by carry logic generating carries in integer multiplication mode. (see Chen col. 4, lines 8-

11: multiplication and addition are performed by large circuits; col. 10, lines 13-36; col. 11, lines 34-40: feedback; first using circuit; then using circuit again with register provided with output from first operational stage (multiplication with feedback)) And, Chen2 discloses wherein to support XOR operations for binary polynomial fields. (see Chen2 col. 4, line 64 - col. 5, line 2; col. 15, lines 29-31: XOR operations)

It would have been obvious to one of ordinary skill in the art to modify Chen to supply a fixed output value in XOR multiplication mode and ensure a result is determined in XOR multiplication mode as taught by Chen2. One of ordinary skill in the art would have been motivated to employ the teachings of Chen2 in order to provide an arithmetic circuit which can perform all arithmetic operations in the finite field, including addition, multiplication, division, exponentiation and inverse multiplication. (see Chen2 col. 3, lines 17-21)

With Regards to Claim 51, Chen discloses the processor as recited in claim 50, wherein the logical circuit is configured to operate as a majority circuit in integer multiplication mode. (see Chen col. 4, lines 8-11: multiplication and addition are performed by large circuits; col. 10, lines 13-36; col. 11, lines 34-40: feedback; first using circuit; then using circuit again with register provided with output from first operational stage (multiplication with feedback)) And, Chen2 discloses wherein to output a zero in XOR multiplication mode. (see Chen2 col. 4, line 64 - col. 5, line 2; col. 15, lines 29-31: XOR operations)

It would have been obvious to one of ordinary skill in the art to modify Chen to

output a zero in XOR multiplication mode as taught by Chen2. One of ordinary skill in the art would have been motivated to employ the teachings of Chen2 in order to provide an arithmetic circuit which can perform all arithmetic operations in the finite field, including addition, multiplication, division, exponentiation and inverse multiplication. (see Chen2 col. 3, lines 17-21)

With Regards to Claim 63, Chen discloses the processor as recited in claim 53, wherein the arithmetic structures are configured to selectively perform one of integer multiplication according to a control signal. (see Chen col. 11, lines 34-40: feedback; first using circuit; then using circuit again with register provided with output from first operational stage; col. 10, lines 13-26: multiple-accumulate instruction; first addend comes from the rightmost k bits of Z register; bits are added to the k bits in the rightmost portion of the product A,B) And, Chen2 discloses wherein to perform XOR multiplication. (see Chen2 col. 4, line 64 - col. 5, line 2; col. 15, lines 29-31: XOR operations)

It would have been obvious to one of ordinary skill in the art to modify Chen to perform XOR multiplication as taught by Chen2. One of ordinary skill in the art would have been motivated to employ the teachings of Chen2 in order to provide an arithmetic circuit which can perform all arithmetic operations in the finite field, including addition, multiplication, division, exponentiation and inverse multiplication. (see Chen2 col. 3, lines 17-21)

With Regards to Claim 64, Chen discloses the processor as recited in claim 63, further comprising a plurality of logic circuits in at least one of the first and second pluralities of arithmetic structures, each logic circuit responsive to the control signal to supply a variable output value in integer multiplication mode, the variable output value varying according to values of inputs supplied to the logic circuit, to thereby ensure a result is unaffected by carry logic generating carries in integer multiplication mode. (see Chen col. 11, lines 34-40: feedback; first using circuit; then using circuit again with register provided with output from first operational stage; col. 10, lines 13-26: multiple-accumulate instruction; first addend comes from the rightmost k bits of Z register; bits are added to the k bits in the rightmost portion of the product A,B) And, Chen2 discloses wherein to supply a fixed output value in XOR multiplication mode and to thereby ensure a result is determined in XOR multiplication mode. (see Chen2 col. 4, line 64 - col. 5, line 2; col. 15, lines 29-31: XOR operations)

It would have been obvious to one of ordinary skill in the art to modify Chen to support XOR operations as taught by Chen2. One of ordinary skill in the art would have been motivated to employ the teachings of Chen2 in order to provide an arithmetic circuit which can perform all arithmetic operations in the finite field, including addition, multiplication, division, exponentiation and inverse multiplication. (see Chen2 col. 3, lines 17-21)

With Regards to Claim 65, Chen discloses the processor as recited in claim 64, wherein the logical circuit is configured to operate as a majority circuit in integer

multiplication mode and to output a zero in the XOR multiplication mode. (see Chen col. 4, lines 8-11: multiplication and addition are performed by large circuits; col. 10, lines 13-36; col. 11, lines 34-40: feedback; first using circuit; then using circuit again with register provided with output from first operational stage (multiplication with feedback)) And, Chen2 discloses wherein to output a zero in the XOR multiplication mode. (see Chen2 col. 4, line 64 - col. 5, line 2; col. 15, lines 29-31: XOR operations)

It would have been obvious to one of ordinary skill in the art to modify Chen and to output a zero in the XOR multiplication mode as taught by Chen2. One of ordinary skill in the art would have been motivated to employ the teachings of Chen2 in order to provide an arithmetic circuit which can perform all arithmetic operations in the finite field, including addition, multiplication, division, exponentiation and inverse multiplication. (see Chen2 col. 3, lines 17-21)

Conclusion

THIS ACTION IS MADE FINAL. Applicant is reminded of the extension of time policy as set forth in 37 CFR 1.136(a).

A shortened statutory period for reply to this final action is set to expire THREE MONTHS from the mailing date of this action. In the event a first reply is filed within TWO MONTHS of the mailing date of this final action and the advisory action is not mailed until after the end of the THREE-MONTH shortened statutory period, then the shortened statutory period will expire on the date the advisory action is mailed, and any extension fee pursuant to 37 CFR 1.136(a) will be calculated from the mailing date of

the advisory action. In no event, however, will the statutory period for reply expire later than SIX MONTHS from the mailing date of this final action.

Any inquiry concerning this communication or earlier communications from the examiner should be directed to Carlton V. Johnson whose telephone number is 571-270-1032. The examiner can normally be reached on Monday thru Friday , 8:00 - 5:00PM EST.

If attempts to reach the examiner by telephone are unsuccessful, the examiner's supervisor, Nasser Moazzami can be reached on 571-272-4195. The fax phone number for the organization where this application or proceeding is assigned is 571-273-8300.

Information regarding the status of an application may be obtained from the Patent Application Information Retrieval (PAIR) system. Status information for published applications may be obtained from either Private PAIR or Public PAIR. Status information for unpublished applications is available through Private PAIR only. For more information about the PAIR system, see http://pair-direct.uspto.gov. Should you have questions on access to the Private PAIR system, contact the Electronic Business Center (EBC) at 866-217-9197 (toll-free). If you would like assistance from a USPTO Customer Service Representative or access to the automated information system, call 800-786-9199 (IN USA OR CANADA) or 571-272-1000.

/Nasser G Moazzami/
Supervisory Patent Examiner, Art Unit 2436

Carlton V. Johnson
Examiner
Art Unit 2436

CVJ
July 4, 2009